Digital Image Watermarking Improvement Using Multiple Duplicated Signals Embedding and Adaptive Retrieval Algorithm Selection Techniques

Komwit Surachat

***Abstract*—In this paper, we propose two techniques to improve the watermark extraction performance and accuracy. The First one is implemented in the embedding process by using a multiple duplicated watermark signals embedding method. The second technique is archived in the retrieval process by using adaptive retrieval algorithm selection to extract the watermark signal. The accuracy of the extracted watermark can be improved by applying these two concepts together. Sets of experiment are fulfilled to testify our proposed technique. The experimental results show the improvement in terms of Normal Correlation (NC) value, compared to the previous methods. The obtained results also perform better in terms of the robustness against various common imaging processing based attacks. Especially in cropping attacks, our proposed watermarking methods show significantly improvement over 0.2 on average in terms of NC compared to the previous proposed methods.**

***Index Terms*—Adaptive retrieval algorithm selection, digital watermarking, multiple duplicated watermark signals embedding.**

I. Introduction

Digital watermarking is a kind of standard technology to maintain access control for the documents. Good introduction on digital watermarking including its essential requirements can be found on [1] and [2].

Currently, many image watermarking methods have been proposed and proved to be robust against various kinds of noises and attacks. Such methods can be classified into frequency and/or spatial domain based watermarking. In the frequency domain, the watermark embedding can be accomplished by modifying the image coefficients from its transformed domain. For instance, Badran et al. [3] proposed the methods to embed the in the Discrete Cosine Transform (DCT) based on image segmentation using Expectation Maximization (EM) algorithm. Also, Patra et al. [4] presented a based Chinese Remainder Theorem (CRT)-based Discrete Cosine Transform (DCT) domain. In addition, [5], [6] and [7] are also proposed the watermarking schemes in frequency domain recently. They presented the method based on applying DCT and DWT to embed and retrieve the watermark signal. They also claimed that the proposed schemes are strongly robust again compression attack. However, many researches demonstrated that the frequency domain based approach was not robust enough against geometrical attack, e.g. cropping. It can survive most image compression standards e.g. JPEG compression standard, though. In contrary, for the spatial domain based approach, it is obvious that the processes of watermark embedding and extraction are simple to perform by modifying the image pixels directly.

For example, M. Kutter et al. [8] presented a method to embed a watermark signal into an image by modifying the pixel using either additive or subtractive depending on the watermark bit, and proportional to the luminance of the embedding pixel. According to their method, the blue colour channel was selected to carry the watermark bit since it is the one that human eye is least sensitive to. Furthermore, the spatial domain watermarking is proposed by using various techniques. For instance, [9] and [10] present the digital watermarking schemes based on independent component analysis and a novel chaos with HVS based, respectively. However, the proposed results are still deficient.

Later, many researchers [11],[12] proposed the methods to improve the quality of the watermarked image by modifying the pixel depending on and proportional to the luminance of the embedding pixel. Especially in [12], T. Amornraksa et al. proposed some techniques to enhance its watermark retrieval performance by balancing the watermark bits around the embedding pixels, tuning the strength of embedding watermark in according with the nearby luminance, and reducing the bias in the prediction of the original image pixel from the surrounding watermarked image pixels. However, all the methods mentioned above encountered a deficiency when implemented with an image having a large number of high frequency components.

Our approach develop a new method based on the previous proposed in [12] by implementing a new embedding technique and a new retrieval algorithm. The descriptions of the proposed are given as follows. The next section gives a brief concept of the digital watermarking based on the modifications of image. Section 3 describes our proposed technique. In section 4, the experimental results are shown and discussed. The conclusion is finally drawn in section 5.

II. Digital Watermarking in Spatial Domain [12]

Firstly, The watermark pixels are converted from {0,1} to {1,-1} by changing the value of the zero bits to be the one bits. Then, the watermark balance and security are improved by using the XOR operation to permute the watermark bits with a pseudo-random bit-stream generated from a key-based stream cipher. The scaling factor s is used to adjust and control the watermark strength of the output previous process

Manuscript received April 25, 2012, revised July 10, 2012.
Komwit Surachat is with Faculty of Science, Prince of Songkla University, Thailand (e-mail: komwit.s@psu.ac.th).

DOI: 10.7763/IJFCC.2012.V2.112

outputs. Then, the embedding process is started by modifying the image pixel in the blue channel $B_{(i,j)}$, in a line scan fashion. The result $B'_{(i,j)}$ are either additive or subtractive, depending on $w_{(i,j)}$, and proportional to the modification of the luminance of the embedding pixel $L_{(i,j)}$. In addition, the modification of luminance $L'_{(i,j)}$ is calculated from a Gaussian pixel weighting mask. The representation of the watermark embedding process can be expressed by

$$B'_{(i,j)} = B_{(i,j)} + w_{(i,j)} s L'_{(i,j)} \quad (1)$$

To extract the watermark signal, the following steps are used to estimate the embedded watermark bit at *(i,j)*. Firstly, each original image pixel in the chosen channel is predicted from its neighboring watermarked image pixels in the same embedding channel. Each original image pixel in the chosen channel is predicted from its neighboring watermarked image pixels in the same channel. The predicted original image pixel $B''_{(i,j)}$ is determined by

$$B''_{(i,j)} = \frac{1}{8}\left(\sum_{m=-1}^{1}\sum_{n=-1}^{1} B'_{(i+m,j+n)} - B'_{(m\_\max,N\_\max)}\right) \quad (2)$$

where $B'_{(m\_max,\, n\_max)}$ is a neighbouring pixel around *(i,j)* that most differs from $B'_{(i,j)}$. Then, the embedded watermark bit $w'_{(i,j)}$ at a given coordinate *(i,j)* can then be determined by the following equation

$$w'_{(i,j)} = B'_{(i,j)} - B''_{(i,j)} \quad (3)$$

where $w'_{(i,j)}$ is the estimation of the embedded watermark w around *(i,j)*. Since $w_{(i,j)}$ can be either 1 and -1, the value of $w'_{(i,j)} = 0$ is set as a threshold, and its sign is used to estimate the value of $w_{(i,j)}$. That is, if $w'_{(i,j)}$ is positive (or negative), $w_{(i,j)}$ is 1 (or -1, respectively). Notice that the magnitude of $w'_{(i,j)}$ reflects a confident level of estimating $w_{(i,j)}$.

III. Proposed Watermarking Methods

In this paper, we proposed two new algorithms to improve the accuracy of the watermark extraction. The first algorithm is implemented in the embedding process and another one is used in retrieval process. Each algorithm detail is given as follows.

A. Multiple Duplicated Signals Embedding Technique

In the embedding process, we proposed a new method to embed a watermark signal into an original host image by sectioning technique. The technique description is given as follows.

The process starts by resizing the watermark logo from the original size 256x256 pixels into 128x128 pixels. Then, the permutation algorithm from the previous method [12] is called for processing the resized signal by using XOR operation with a pseudo-random bit-stream generated from a key-based stream cipher to improve the balance of the watermark bits and security of the embedding process.

After the signal is permuted, the obtained signal is then sectioned into 4 same size windows (64x64 pixels) and named all window with letter a, b, c and d.

In this step, the 256x256 embedding pane must be generated by creating 4 more windows. Then, we must used all 5 windows (one is original window and other 4 window is come from this step) to create the 256x256 pane. Note that, one window must be used as a shared window with other 4 windows (the original window is selected as a shared window). In this situation, the order of each window will be re-arranged follow the shared window. In addition, we set up a condition of re-rearranging process that is only the same labeled part can be arranged in the same position. Thus, the re-arranging algorithm will find out the appropriate solution embedding pane.

After we obtain the embedding pane, we tune it up with signal scaling factor s for adjusting the strength signal. The embedding process is then started by adding the original host image with the tuned signal.

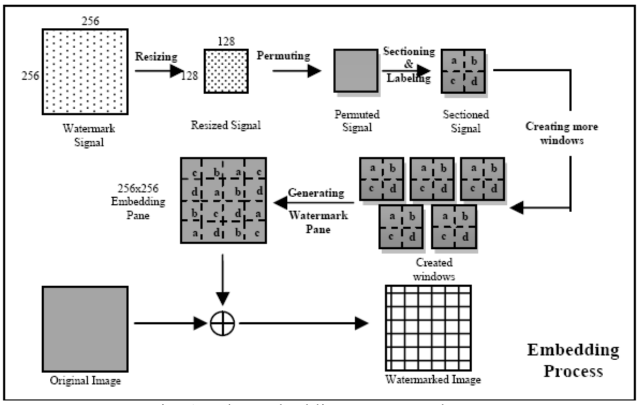


Fig. 1. The embedding process scheme.

In the retrieval process, the inverse operation of the embedding process has been applied by these following steps.

We use the new proposed prediction technique (as explain in section B.) to generate a predicted original image. The extracted watermark signal is then determined by subtraction between the watermarked image and the predicted image following equation (3).

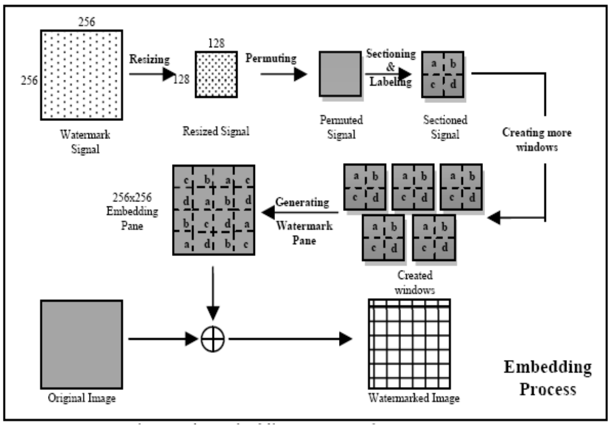


Fig. 2. The retrieval process scheme.

The extracted watermark signal is re-sectioned into 5 parts similar to pre-processing process (Inverse cycle). After sectioning, the order of each part is not arranged into the right order. For instance, the order of part No.1 is c, b, d and a. Thus, the re-ordering operation re-arrange the order into the right one (a, b, c and d). Conceptually, all parts are re-arranged correctly.

After all parts are re-arranged into the right order. The obtained results must be voted to only one signal. In this situation, each window contains watermark bits (value ∈ {1, 0}). The majority voting decides a final bit from all of bits from each window. For example, there are {1, 1, 0, 0, and 0} in voting list. In this case, the majority bit should be 0 because there are 3 zero bits and only 2 one bits.

The final process is resizing size back to 256x256 same as the original watermark signal.

B. Adaptive Retrieval Algorithm Selection Technique

Initially, we analyzed the influenced factors that affect the watermark extraction accuracy. From equation (2), it can be seen that the accuracy of an original pixel prediction mainly depends on its nearby pixel values. In addition, a higher prediction performance of the original pixel comes from a small difference between the nearby pixels. On the other hand, a large variation in the neighbor pixels can cause bias in the prediction process that results an erroneous original predicted pixel value. However, this error can be minimized by eliminating the surrounding pixels around position *(i,j)* that contain too high difference value from the remaining pixels. Conceptually, if all pixels in the prediction area are not too much different, there is no any pixel that will be eliminated from the prediction process. By the first assumption of the previous method in [12], any pixel value within an image is close to its surrounding neighbors, so that a pixel value at given coordinate *(i,j)* can be predicted by averaging all of its neighbor pixels.

In contrary, in general prediction case of any watermarked image, it cannot be possible that the first assumption will hold all of the cases. There are so many prediction areas which are out of the case of the assumption. Therefore, we proposed a new technique to reduce the error from the prediction process by using adaptive algorithm selection. We provided three algorithms to support our idea based on statistical theory. The Standard Deviation (*SD*) value is use to be a criterion for choosing a suitable algorithm in the selected prediction area.

Base on our concept, we categorized the prediction cases into 3 cases those are small, medium and large standard deviation contained area. In the first case, we apply the first assumption to estimate the pixel value by averaging all eight surrounding pixels. If the calculation of *SD* value from selected window is very low that means the first assumption is held, therefore the averaging pixels method work strongly in this situation. In contrast, if the *SD* value is quite or extremely high, the prediction error will subsequently occur.

Accordingly, two additional algorithms are proposed to solve this problem. We proposed to eliminate two bias pixels in medium *SD* case by sorting all pixel values in prediction area and removing the first and last pixels from the sorted pixels. The removed pixels are guaranteed to be one positive watermark bit and one negative watermarking by this method.

The last provided algorithm is the majority elimination voting algorithm. This algorithm is for used in case of high *SD* value only. If the *SD* value of the selected area is in this case, the algorithm will be selected. The first calculation is simple computing by averaging all eight pixel values. Then, the second calculation is produced by eliminating the first and last pixels in the array before calculating the average of the remaining. Conceptually, the third calculation is computed by removing more two pixels from the sorted pixels. All results of three calculations are then subtracted by central pixel and kept only the sign of the calculations. Consequently, the sign voting majority are used to consider the final result. The voting machine will vote the positive or negative sign ups to the result and then it will return one prediction value of the majority for the next process.

The pseudo code of our proposed retrieval watermarking process is given below.

Function Adaptive Retrieval Algorithm Selection Method at coordinate *(i,j)*

```
neighbor [] – array of surrounding pixels
cen – central pixel in prediction window
SD – standard deviation of neighbor []
size – size of surrounding pixels in calculation
C1 – criterion of small standard deviation value
C2 – criterion of medium standard deviation value
sign [] – array of calculation sign

Function Start

if (SD <= C1)
    goto Algorithm I
elseif (C1<SD<C2)
    goto Algorithm II
else
    goto Algorithm III
endif

Algorithm I : Pixel Averaging
    Return (mean(neighbor[])

Algorithm II :  Min-Max Pixels Eliminating
    temp [] = sort(neighbor[])
    temp [] = temp [2:7]
    Return  (mean(temp[]))

Algorithm III :  Majority Voting
    temp [] = sort(neighbor[])
    sign(1) = cen – mean(temp[])
   sign(2) = cen – mean(temp[2:7])
   sign(3) = cen – mean(temp[3:6])
            Return (vote(sign[]))

END
```

IV. The Experimental Results and Analysis

A. The Experimental Settings

In the experiments, nine 256x256 pixels standard color images were used as the original images, namely *'Lena'*, *'Tower'*, *'House'*, *'Fish'*, *'Bird'*, *'Pens'*, *'Flowers'*, *'Pepper'* and *'Baboon'* . Also, the 256x256 pixels black and white image containing logo *'PSU'* was used as a watermark signal.

(a) *Lena* (b) *Pens* (c) *Flowers* (d) *PSU ICT*

Fig. 3. The original testing images and the watermark logo.

B. Standard Deviation Criterion Evaluation for Algorithm Selection

The first set of experiments was performed to identify the optimum criterion used to decide what an algorithm will be selected for calculation. In this experiment, we evaluated the watermark extraction accuracy at various computed standard deviation (*SD*) by comparing the average *NC* values obtained from different criterion derived from σ of the prediction window. We suggested that it should categorize the prediction cases into 3 types based on σ of each prediction window. The first criterion was defined by varying σ from 0 to 20 for the small standard deviation case. The best result came from $\sigma = 12$, we thus set $\sigma = 12$ as a criterion of the first case. Then, we justified the second criterion by varying period from 12 to 40. The best averaged result was from $12 < \sigma \leq 30$. We then set this period as a second criterion. Consequently, the third criterion was automatic set as $\sigma > 30$. The results are illustrated in Fig. 4.

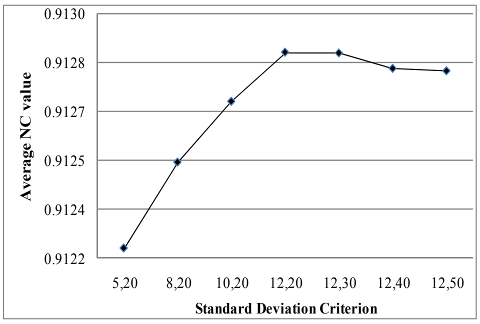


Fig. 4. The standard deviation criterion evaluation.

Next, we computed and compared the performance of algorithms by calculating the averaged *NC* value of all extracted testing images. We tested each algorithm separately i.e. we define the averaging all eight neighbor pixels as algorithm I, the min-max value elimination as algorithm II, the majority voting elimination as algorithm III and the proposed method as algorithm IV. As illustrated in Fig.5, the performance of algorithm I, II, II and IV are 0.9107, 0.9122, 0.9122 and 0.9128, respectively. That means the proposed algorithm archived the highest performance in terms of *NC* value. The results also supported our proposed idea.

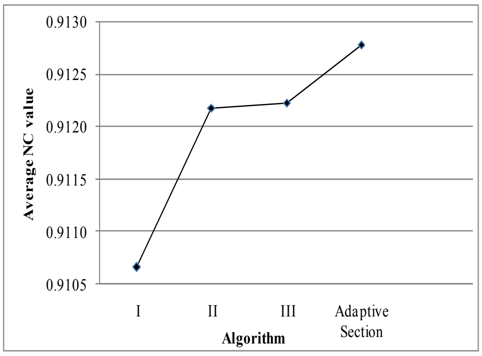


Fig. 5. The performance comparison of each algorithm.

Then, we evaluated times of calculation of each algorithm. As shown in TABLE I, the frequency of function call of the first algorithm is the highest. We found that more than 50% of all areas in the images are considered in the 1st case, 30% in 2nd case and 20% in 3rd case, respectively. Conceptually, we know that almost all of general images mainly contain low frequency area almost entire of the image and just a few of image edge in only some part. On the other hand, if the image is a high frequency image, high frequency area will be more than low frequency area. Thus, algorithm II will be called more than other algorithms in case of *'Fish'* as shown in the table.

TABLE I: THE TIMES OF CALCULATION OF EACH ALGORITHM

Images	Times of Calculation		
	Algorithm I	**Algorithm II**	**Algorithm III**
Lena	44920	16893	2703
Bird	37939	17496	9081
Fish	728	10974	52814
Tower	40304	17261	6951
Baboon	4158	33497	26861
House	51344	8899	4273
Pens	30122	26796	7598
Peppers	43001	17267	4248
Flowers	32400	22731	9385
Average	**31657.33**	**19090.44**	**13768.22**

C. Performance Comparison

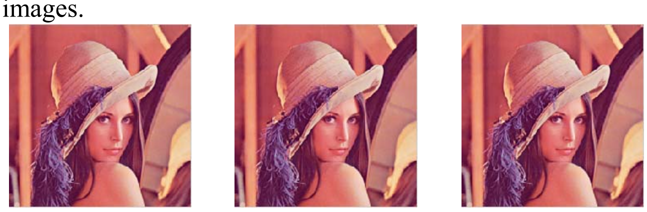
In this experiment, we evaluate the performance of our proposed watermarking method. The result were then presented and compared to the previous method in [12]. Note that, in this experiment the *NC* value obtained from each method at various signal strength was measured at the equivalent image quality, at *PSNR* ≈ 35 dB with the difference of less than 0.0001 *dB*, in order to archive a fair comparison.

It can be obviously seen that the performance of our proposed method outperformed the previous method in [12], judge from the highest *NC* value at all fixed *PSRN* value. As shown in the table below, the improvements of our proposed method were about 0.049, 0.067 and 0.077 at fixed *PSNR* 30, 35 and 40, respectively.

TABLE II: AVERAGE *NC* VALUES AT DIFFERENT FIXED PSNR

PSNR	Average *NC* value		
	Previous Method in [12]	**Proposed**	**Improvement**
30.0000	0.8865	0.9356	**0.0491**
35.0000	0.8457	0.9128	**0.0671**
40.0000	0.8026	0.8791	**0.0765**

Moreover, Fig. 6 illustrated different versions of watermark images '*Lena*' both our proposed and the previous method. From the figure, we can difficultly notice the difference between the original image and the watermark images. From our observation, the quality of the watermarked images was undoubtedly comparable to the host images.



(a) Original image (b) Previous method (c) Proposed method

Fig. 5. The resultant watermarked image '*Lena*'.

D. Robust Against Attacks

Finally, the robustness of the proposed watermarking method was evaluated by applying six different types of attack. The NC values from the attacked images were then computed and compared. A list of the attacks in the experiment consisted of additive Gaussian distributed noise with zero mean at various variances, the cropping attacks at various percentage and various cropping styles, the salt and pepper noise at various densities, the blurring attack at theta = 11 and various length and the contrast adjustment attack at various contrast scaling factors.

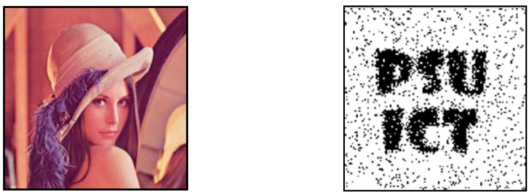
The results of the experiment are shown in TABLE III. Also, the improvements in terms of retrieval accuracy are presented in the last column.

TABLE III: AVERAGE NC VALUES AT DIFFERENT TYPES OF ATTACK

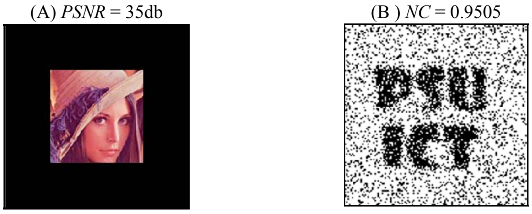
Attack Types	**Average *NC* Values**		
	Previous Method in [12]	**Proposed Method**	**Improvement**
Non-attacked *PSNR* = 35dB	0.8457	0.9128	0.0671
Non-attacked *PSNR* = 40dB	0.8026	0.8791	0.0765
Gaussian Noise Variance = 0.01	0.7219	0.7982	0.0763
Cropping Percentage = 50	0.6986	0.8720	**0.1735**
Cropping to Right Corner 50%	0.7637	0.8779	**0.1142**
Salt and Pepper Noise density = 0.06	0.8370	0.8934	0.0564
Blurring Theta =11 Len =8	0.7162	0.7954	0.0792
Contrast Adjustment 50%	0.8179	0.8843	0.0663

As shown in TABLE III, our proposed method performed better in every types of attack including non-attacked image. Especially in the cropping attack, at cropping percentage = 50 the average NC value was explicitly better than the previous method about 0.2 and the extracted watermark image was still clearly readable by human eyes as shown in figure 2(D)-2(E). Moreover, in the non-attacked cases, from figure 1(A) we hardly detect the difference between the original image and the watermarked image. From our observation, the quality of the watermarked image was undoubtedly comparable to the original host image.

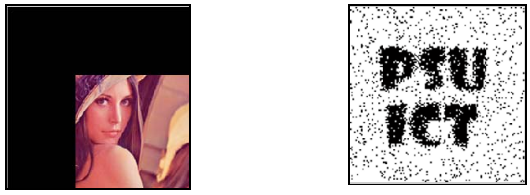
In addition, the resultant extracted watermark images are shown in Fig. 6. Accordingly, we proposed a multiple duplicated embedding technique in the embedding process by inserting same signals in five different positions, the robustness against cropping attack should be improved. As a result, in Fig. 6 (B) even though some parts of the images are missing, the watermark signal can be still recovered back and readable with human eyes. Also in the other attack types, the performances of proposed algorithm are improved in terms of *NC* values and readability with human eye test.



(A) *PSNR* = 35db (B) *NC* = 0.9505



(B) Cropping 20% (C) *NC* = 0.9088



(D) Cropping 50% (E) *NC* = 0.9505

Fig. 6. The attacked image '*Lena*' and the resultant extracted watermark image '*ICT PSU*'.

V. CONCLUSION

In this paper, we have proposed two new methods to improve the accuracy of the extracted watermark by using Multiple Duplicated Signals Embedding and Adaptive Retrieval Algorithm Selection Techniques. We have proposed to section the watermark image before embedding in to the original host image. In addition, in the extraction process, the Adaptive Retrieval Algorithm Selection Technique is presented to extract the watermark signal by using three algorithm based on Standard Deviation value of prediction area. The experimental results have shown the improvement in every attack types including the non-attacked image.

REFERENCES

[1] I. J. Cox. L. Mathew, M. Miller, A. Bloom, J. Fridrich, and T. Kaller, "Digital Watermarking and Steganography," *Morgan Kaufmann. Los Altos*, CA. USA. 2002, ISBN: 1-55860714-5.

[2] F.Y. Shih, "Digital Watermarking and Steganography Fundamental and Techniques," *CRC Press*, 2007.

[3] E. F. Badran, A. Ghobashy, and K. E. Shenawy. "DCT-Based Digital Image Watermarking Via Image Segmentation Techniques," in *Proc. of ITI 4th International Conference on Information and Communications Technology*, Alaska. USA. 2006.

[4] J. C. Patra, A. K. Kishore, and C. Bornand, "Improved CRT-based DCT domain watermarking technique with robustness against JPEG compression for digital media authentication," in *Proc. of 2011 IEEE International Conference on Systems*, Man, and Cybernetics, 2011, pp. 2940 – 2945.

[5] Z. Li, "A new blind robust image watermarking scheme in SVD-DCT composite domain," *2011 18th IEEE International Conference on Image Processing (ICIP)*, 2011, pp. 2757 – 2760.

[6] Q. Li. "Adaptive DWT-SVD Domain Image Watermarking Using Human Visual Model," *The 9th International Conference on Advanced Communication Technology*, 2007, pp. 1947 – 1951.

[7] K. A. Navas, "DWT-DCT-SVD based watermarking," *3rd International Conference on Communication Systems Software and Middleware and Workshops*, 2008, pp. 1947 – 1951.

[8] A. Hajisami. "Watermarking Based on Independent Component Analysis in Spatial Domain," *2011 UkSim 13th International Conference on Computer Modelling and Simulation (UKSim)*, 2011,

[9] L. Gang. "A novel chaos and HVS based image watermarking algorithm," *2010 International Conference on Computer, Mechatronics, Control and Electronic Engineering (CMCE)*, 2010

[10] M. Kutter, F. Jordan, and F. Bossen. "Improved Digital Signature of Colour Images using Amplitude Modulation," *Journal of Electronic Imaging*, 1998, pp. 326 – 332.

[11] R. Puertpun and T. Amornraksa, "Pixel Weighting Marks in amplitude Modulation of Colour Image Watermarking," in *Proceedings of the IEEE ISSPA*. Kuala-Lumpur. 2001, pp. 194 – 197.

[12] T. Amornaksa and K. Janthawongwilai, "Enhanced Images Watermarking Based on Amplitude Modulation," *Journal of Image and Vision Computing*, 2006, pp. 111 – 119.